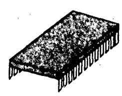
Nonvolatile, in-circuit-reprogrammable memories

Vendors of nonvolatile, in-circuit-reprogrammable, semiconductor memories want you to replace electromechanical memory components, such as DIP switches and disk drives, with chips. To spur your move toward silicon, IC vendors are rapidly expanding nonvolatile memory capacities beyond 1M bits and slashing the cost per bit.



Steven H Leibson,
Senior Regional Editor

Computer designers abandoned core memory in the early 1970s because silicon memory (RAMs and EPROMs) offered equal or better performance for less money. However, system designers also lost something in the bargain. Semiconductor RAMs furnish volatile memory; they remember only as long as they receive power. EPROMs need no power to retain their contents but you can't reprogram an EPROM without first erasing it for 20 minutes under a UV lamp. Consequently, disk drives now fill the nonvolatile memory needs of many computer systems.

Unfortunately, disk drives also have liabilities. They are much larger and heavier than ICs, they're more easily damaged and slower than semiconductor memory, they require a relatively complex disk controller, and they consume a lot of power. Disk drives may make no sense at all in small, portable, or embedded µP-based systems. Recognizing the inherent limitations of EPROMs and disk drives for nonvolatile storage, the makers of parallel-access, in-circuit-reprogrammable, nonvolatile memory ICs plan to restore nonvolatility to your processor's main memory, as in the days of core. These vendors are pushing device capacities beyond 1M bits (to 16M bits for hybrid modules) and dropping costs to $10/megabit as a means of swaying you to their way of thinking.

When you decide to use some form of nonvolatile memory IC in your next design, you immediately confront a range of technology choices. You can pick from battery-backed static RAMs (SRAMs), nonvolatile RAMs, EEPROMs, and flash EEPROMs. Your choice will depend on several factors including how much nonvolatile memory you need, how often you need to change the nonvolatile memory's contents, and how much you're willing to spend. The selection guide in **Table 1** can help you choose the type of device that best suits your needs. Choosing a particular device within a given device type then becomes a somewhat simpler problem. Unfortunately, none of the nonvolatile-memory technologies listed in **Table 1** offers the ideal nonvolatile-memory characteristics of unlimited endurance, fast storage times, and low cost. All exhibit some form of wearout failure and all require that you make some compromises.

If you need nonvolatile storage that closely mimics the infinite read/write characteristics of RAM,

Order Number: 295062-001

then battery-backed SRAMs may well be your best choice. These devices work just like SRAM because they're built with SRAM chips. In addition to the RAMs, these devices incorporate one or two batteries and a power-management circuit that switches over from system power to battery power when necessary.

Table 2 lists battery-backed SRAMs offered by two vendors: Dallas Semiconductor and SGS-Thomson. These companies offer products whose capacities range from 2k to 128k bytes. Some of the products listed also incorporate an electronic time-of-day clock and calendar. The products' integral battery allows the clock and calendar to keep track of the time and date even when a system is turned off.

However, the integral battery is not just an asset, it is also the technology's main liability. Batteries die. **Table 2** lists the minimum number of years you can expect the battery to last for each device. Most of these products retain data for about a decade. After the battery is drained, the SRAM will operate as a RAM, but the product's nonvolatile abilities disappear.

The battery-backed SRAM's integral battery is not only an asset, it is also the technology's main liability.

The batteries in battery-backed SRAMs cannot be recharged, so you must replace the entire device to restore nonvolatile memory to your system. If you prefer to provide your own battery (perhaps one that's rechargeable), SGS-Thomson offers the MK48C02, which is a 2k-byte SRAM with integrated power management. The device has separate battery pins, which allow you to connect a backup power source to the RAM in addition to the normal power supply. The 48C02 costs less than $10 (1000).

Other types of nonvolatile semiconductor memories based on floating-gate electron storage (nonvolatile RAMs, EEPROMs, and flash EEPROMS) feature much longer data retention than battery-backed SRAMs. Although most of the data sheets for these floating-gate devices claim 10-year data retention, the technology is actually much better. In fact, Xicor's nonvolatile RAM and EEPROM data sheets specify 100-year retention times. Most nonvolatile-memory vendors will readily negotiate longer retention-time specs if your application requires them. Apparently, many of these vendors don't want you to think that you're paying extra for the longer retention specs so they use the de facto "industry standard" spec of 10 years.

You should also be aware of the difference in retention-time specifications between floating-gate memory products and battery-backed SRAMs. The retention-time clock for floating-gate memories starts when you store data in the IC. For battery-backed memories, the retention-time spec refers to the elapsed time since the device was first powered up.

If the limited life of a battery concerns you, but you still need infinite read/write capabilites, consider using nonvolatile RAM for your nonvolatile memory. By merging SRAM and EEPROM arrays on one chip, nonvolatile RAMs do not require a battery and provide many of the same advantages provided by battery-backed SRAMs. While the system supplies power, the nonvolatile RAM's SRAM array pro-

Table 1—Nonvolatile-memory characteristics

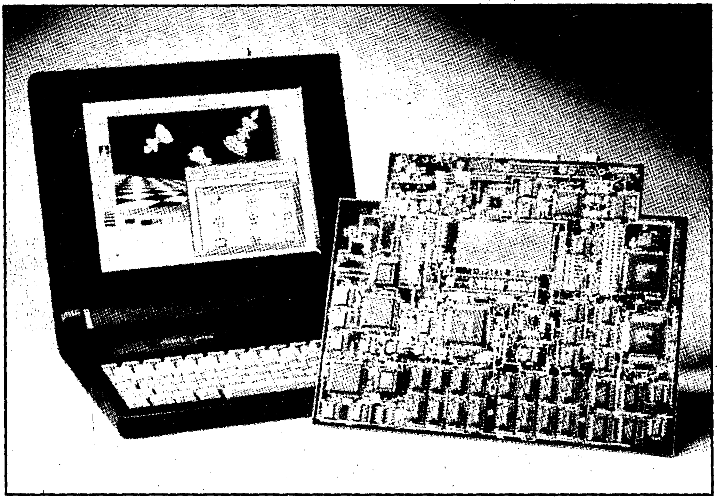
	Nonvolatile-memory technology				
Characteristic	**Battery-backed static RAM**	**Nonvolatile static RAM**	**Full-featured EEPROM**	**Flash EEPROM**	**EPROM**
Write granularity	1 byte	1 byte (RAM), entire device (EEPROM)	1 byte or 1 page	1 byte or 1 page	1 byte
Rewrite granularity	1 byte	1 byte (RAM), entire device (EEPROM)	1 byte or 1 page	Entire device (Note 1)	Entire device (Note 2)
Relative storage speed	Very fast	Very fast (RAM), slow (EEPROM)	Slow (1 byte), medium (1 page)	Slow (1 byte), medium (1 page)	Slow
Programming and erase power requirments	5V	5V	5V	5V, or 5V and 12V	Not reprogrammable in system
Write endurance (cycles)	Unlimited	Unlimited (RAM) 10,000 to 100,000 (EEPROM)	10,000 to 100,000	100 to 10,000	<1000
Bit capacity per device	16k to 1M	256 to 64k	4k to 1M	256k to 2M	8k to 4M
Relative cost per bit	Highest	High	Medium	Low	Lowest

Notes: 1. Sector-erasable flash EEPROMs have 1-sector rewrite granularity.
2. EPROMS cannot be erased electrically. They must be erased with a UV lamp.

vides unlimited read/write capabilities. At any time, you can transfer the entire contents of the RAM array to the EEPROM array with one command. You can also transfer information stored in the nonvolatile RAM's EEPROM array to its RAM array with one command. In addition, nonvolatile RAMs automatically transfer their EEPROM's contents to their RAM array when power is applied.

Is 27 years long enough?

Like any floating-gate memory technology, the nonvolatile RAM's EEPROM array accepts only a limited number of changes before it will fail. However, as **Table 3** shows, that limit is at least 10,000 storage cycles. If you only transfer the RAM array to the EEPROM array when shutting the system down, that limitation should not represent much of a problem. For a product that is switched off once a day, the 10,000-cycle nonvolatile RAMs should last more than 27 years.



To hold the BIOS in its VH-286 notebook PC, Airus Computer Corp (Chicago, IL) uses a flash EEPROM from Seeq, which allows the company to change its BIOS over the telephone via a modem.

Unfortunately, if you use the nonvolatile RAM to save data prior to the loss of system power, you'll need circuitry that can foresee the power loss sufficiently early enough to complete the data transfer from RAM to EEPROM. Such predictive circuitry can be tough or impossible to design, so you may need to initiate transfers more frequently in case the system's power vanishes unexpectedly. If so, 10,000-cycle and even 100,000-cycle endurance may become a real limitation. However, if you only need to store infrequently updated items, such as con-

Table 2—Representative battery-backed static RAM modules

Manufacturer	Model	Size (bits)	Organ-ization (bytes)	Access time range (nsec)	Minimum battery life (years)	Cost ($) (1000)	Comments
Dallas Semiconductor	DS1220	16k	2k	150 to 200	10	6.25 to 7.20	Industrial-temperature version available
	DS1225	64k	8k	70 to 200	10	8.75 to 9.10	Industrial-temperature version available
	DS1230	256k	32k	100 to 200	10	18.15 to 22.50	
	DS1243	64k	8k	200	10	16.90	Real-time clock/calendar
	DS1245	1M	128k	100 to 120	10	58.75 to 61.25	
	DS1387	32818	4k+50	370	10	13.75	Real-time clock/calendar, plugs into IBM PC/AT clock socket
	DS2217	1M+parity	128k+1 parity bit	200	10	70	30-pin, single-in-line package
	DS2230	512k	64k	200	10	39.60	Dual-ported device: one 8-bit port and one serial port, 40-pin single-in-line package
SGS-Thomson	MK48T02	16k	2k	120 to 250	11	17.10 to 27.12	Real-time clock/calendar
	MK48Z02	16k	2k	120 to 250	11	7.50 to 12.45	Industrial-temperature version available
	MK48T08	64k	8k	150 to 250	10	22.50 to 25.50	Real-time clock/calendar
	MK48Z08	64k	8k	70 to 200	11	13.13 to 18	
	MK48Z30	256k	32k	100 to 120	10	27 to 31.50	Available the second quarter of 1991
	MK48T87	512	64	240	10	11.25	Real-time clock/calendar, multiplexed address/data bus

figuration information or calibration data, the nonvolatile RAM may be a perfect choice.

Nonvolatile RAMs meld the infinite read/write capabilities of RAM with the nonvolatile storage of EEPROM. This combination offers more flexibility than other types of nonvolatile semiconductor memory, but you must pay for this flexibility. Smaller nonvolatile RAMs only cost a few dollars, so they are frugal replacements for DIP switches, jumpers, and other mechanical contrivances often used to store small amounts of semipermanent data on a board. However, the nonvolatile RAM's relatively high cost per bit makes it unattractive for large storage requirements. Further, the inherent complexity of the nonvolatile RAM's combined static-RAM/EEPROM cell limits the amount of memory that an IC vendor can fabricate on one IC using nonvolatile RAM cells. For these two reasons, you can buy only relatively small nonvolatile RAMs. Simtek's 16k- and 64k-bit parts offer significantly more storage space than nonvolatile RAMs from other vendors. Even so, 8k bytes doesn't seem to go as far these days as it once did.

For a product that is switched off once a day, the 10,000-cycle nonvolatile RAMs should last more than 27 years.

If you can restructure your design requirements so that you don't need unlimited read/write capabilities, you can conceptually eliminate the nonvolatile RAM's SRAM. You then end up with a part that is all EEPROM. Full-featured EEPROMs offer more capacity at a lower cost per bit than nonvolatile RAMs. **Table** 4 lists a large sample of available full-featured EEPROMs. (Note: **Table** 4 lists only bit-parallel EEPROMS. EDN plans to cover serial EEPROMs later this year.)

Restructuring your design requirements can be easier than you might think because a full-featured EEPROM differs from an EEPROM array of a nonvolatile RAM in one key aspect: You can change individual storage locations in a full-featured EEPROM independently. Consequently, you can write to one storage cell of a full-featured EEPROM several thousand times without reducing the endurance of the device's other memory locations. So, if your application changes stored information frequently, but doesn't change the same locations each time, the relatively limited endurance of full-featured EEPROMs may pose no problems. Alternatively, if you don't need to use an EEPROM's full capacity, you can adopt a scheme that uses only part of the EEPROM's memory until you near the endurance limits and then switches to a different section of the device. One of the EEPROM's memory locations can serve as a pointer to the section currently in use.

Nonvolatile RAMs can be faster than EEPROMs for multiple-byte

4

Table 3—Representative nonvolatile static RAMs

Manufacturer	Model	Size (bits)	Organization (bytes)	Access time range (nsec)	Store cycle time (entire array) (msec)	Array endurance (store cycles)	Cost ($) (1000)	Comments
Catalyst	CAT22C10	256	64x4 bits	200 to 300	10	10,000	3 (10,000)	Industrial-temperature version available
	CAT22C12	1024	256x4	200 to 300	10	10,000	3.75 (10,000)	Industrial-temperature version available
Seiko Instruments	S-2210R	256	64x4 bits	300	10	10,000	3.03	Industrial-temperature version available
	S-2212R	1024	256x4 bits	300	10	10,000	4.95	Industrial-temperature version available
Simtek	10C48	16k	2k	25 to 55	10	10,000	8.13 to 26.72	Pin-driven EEPROM store/recall, industrial- and military-temperature versions available
	10C68	64k	8k	25 to 45	10	10,000	18.87 to 59.37	Pin-driven EEPROM store/recall, industrial- and military-temperature versions available
	11C48	16k	2k	25 to 55	10	10,000	10.19 to 32.06	Software-driven EEPROM store/recall, industrial- and military-temperature versions available
	11C68	64k	8k	25 to 45	10	10,000	18.87 to 59.37	Software-driven EEPROM store/recall, industrial- and military-temperature versions available
Xicor	X2001P	1k	128	200 to 300	10	100,000	5.16	Industrial-temperature version available
	X2004P	4k	512	200 to 300	10	100,000	11.82	Industrial-temperature version available
	X2210P	256	64x4 bits	300	10	100,000	2.99	Industrial-temperature version available
	X2212P	1k	256x4 bits	300	10	100,000	4.92	Industrial-temperature version available

storage operations, but you can overcome this problem as well. Because the nonvolatile RAM fills its entire EEPROM array in one storage cycle, it has a high "effective storage rate" per byte. Some data-acquisition and real-time processing applications that use nonvolatile storage to record data at periodic intervals can't tolerate the milliseconds-long intervals between each store operation if the storage time exceeds the sampling interval. Nonvolatile RAMs circumvent this problem by allowing the application to accumulate data in the IC's RAM array and then store all of the accumulated data in one operation.

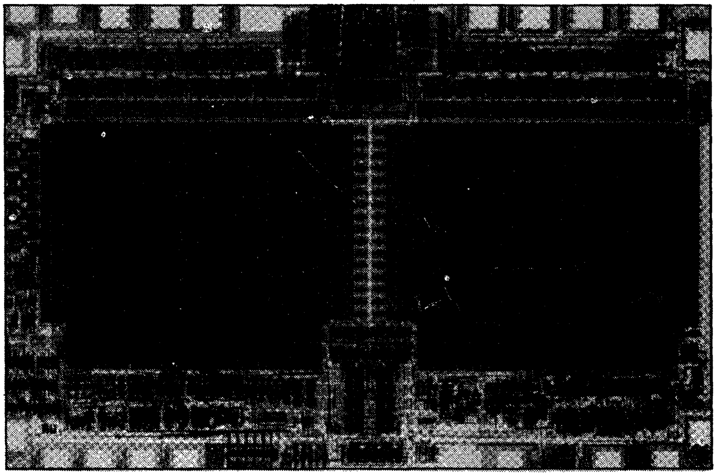


Fig 1—Two EEPROM sections allow this Xicor XC88C64 to store data and provide read capability simultaneously. This feature may allow you to reduce the number of memory devices in your design to one.

Pages save storage time

Some full-featured EEPROMs have page buffers that can store a contiguous page of memory locations in one operation. EEPROM page sizes range from 32 to 128 bytes. Using the page-write feature, you can accumulate a page worth of data and then store that page in the EEPROM in about the same time required to store 1 byte. The full-featured EEPROM's page buffer, therefore, provides advantages similar to those of the nonvolatile RAM's RAM array.

This page-write feature works similarly for all vendors' full-featured EEPROMs that have page buffers. The EEPROM senses the first write cycle and starts an internal timer. If another write cycle occurs before the timer reaches a predetermined value (usually 100 μsec), the additional byte enters the page buffer, the timer resets, and the EEPROM postpones the actual storing of the information in the EEPROM array. Thus you can quickly fill the EEPROM's page buffer without activating the milliseconds-long storage cycle.

When you stop writing to the page buffer, the timer times out and the EEPROM stores the entire page in one operation. This scheme effectively reduces the time required to store an individual byte by the reciprocal of the page size. For example, if an EEPROM stores 64 bytes from its page buffer in one cycle, the "effective" time required to store 1 byte is $1/64$ of the page-write storage time.

You can't use just one

Most full-featured EEPROMs have another shortcoming that complicates their use: They don't respond to read cycles while performing a store operation; the EEPROM's contents become inaccessible for several msecs at a time. If you store executable code in an EEPROM with this trait, your processor will not be able to fetch instructions from the EEPROM while it is storing information. Consequently, you cannot use just a full-featured EEPROM to store your code in systems that modify the code space because access to the code will be interrupted for long periods of time. You can solve this problem by combining an EPROM and an EEPROM, or two EEPROMs. If you use two EEPROMs, be certain to initiate store operations in only one device at a time. You can also solve this problem using one of Xicor's X88C64s, which incorporates two independent EEPROM arrays (**Fig 1**). This device allows access to one of its EEPROM arrays while it is storing data in the other.

The X88C64 represents the latest step in the evolution of full-featured EEPROM design. Early EEPROMs were hard to use because they incorporated no address or data latches to hold values during the long store operations, they required high programming voltages, they were sensitive to power-supply transitions when a system was powered up or down, and they required external timing circuitry or software to control the store cycle. As the IC vendors developed expertise with EEPROM technology, they eliminated these design obstacles by adding circuitry to the EEPROMs.

However, the circuitry added for each new feature also adds cost to the device. Because of their cost per bit, full-featured EEPROMs pose no great threat to disk drives. By paring the full-featured EEPROM to the barest essentials, many ven-

dors of full-featured EEPROMs now offer a relatively new type of device, the flash EEPROM, which will give both EPROMs and disk drives some competition. **Table 5** lists flash EEPROMs offered by some sources.

To make flash EEPROMs inexpensive, some IC designers remove all circuitry that isn't absolutely necessary. You get a less expensive memory IC, but you also assume more responsibility for programming and erasing the device. For example, many flash EEPROMs lack on-chip charge pumps and require an external 12V source for programming and erasing operations. Full-featured EEPROMs and flash devices from Atmel, Catalyst, and Texas Instruments generate

Table 4—Representative full-featured EEPROMs

Manufacturer	Model	Size (bits)	Organization (bytes)	Read/access time range (nsec)	Max write time (msec)	Page size (bytes)	Endurance (write cycles)	Cost ($) (1000)	Comments
Atmel	AT28C04	4k	512	150 to 250	1	NA	10,000	3.20 to 6.50	0.2-msec fast-write and 100,000-cycle versions available
	AT28C16	16k	2k	150 to 250	1	NA	10,000	4.50 to 14	0.2-msec fast-write, 100,000-cycle, 45-nsec, and low-power versions available
	AT28C64	64k	8k	150 to 250	1	NA	10,000	6.50 to 17	0.2-msec fast-write, 100,000-cycle, 55-nsec, and low-power versions available
	AT28PC64	64k	8k	150 to 350	2	32	10,000	6.50 to 17	100,000-cycle version available
	AT28C256	256k	32k	150 to 350	10	64	10,000	40 to 59	3-msec fast-write and 100,000-cycle versions available
	AT28C010	1M	128k	120 to 250	10	128	10,000	350 to 550	100,000-cycle version available
	AT28C1024	1M	64k×16 bits	120 to 250	5 (typ)	64 words	10,000	420 to 650	100,000-cycle version available
Catalyst	CAT28C16A	16k	2k	200	10	NA	10,000	3 (10,000)	
	CAT28C16V3	16k	2k	700	20	NA	10,000	3.75 (10,000)	Operates with power supplies as low as 3V
	CAT28C64A	64k	8k	120 to 200	10	32	10,000	5 (10,000)	Industrial-temperature version available
	CAT28C256	256k	32k	200 to 300	10	64	10,000	25 (10,000)	
Dense-Pac Microsystems	DPE256Q8	2M	256k	70 to 250	10	64	10,000	380 to 760	Industrial- and military-temperature versions available
	DPE256S8N	2M	256k	135 to 300	10	128	10,000	1075 to 2150	Industrial- and military-temperature versions available
	DPE3232V	1M	128k×8, 64k×16, or 32k×32 bits	70 to 250	10	64	10,000	200 to 400	Industrial- and military-temperature versions available
	DPE32X16A	512k	64k×8 or 32k×16 bits	70 to 250	10	64	10,000	120 to 1250	Industrial- and military-temperature versions available
	DPE41288	1M	128k	90 to 250	10	64	10,000	220 to 450	Industrial- and military-temperature versions available
	DPE45128	4M	512k	100 to 350	10	64	10,000	800 to 1600	Industrial- and military-temperature versions available
	DPE6434	2M	256k×8, 128k×16, or 64k×32 bits	90 to 250	10	64	10,000	400 to 800	Industrial- and military-temperature versions available
	DPE832V	55 to 250	32k×8, 16k×16, or 8k×32 bits	55 to 250	10	32	10,000	140 to 275	Industrial- and military-temperature versions available, pin-grid-array package
	DPE8M624	90 to 250	128k×8 or 64k×16 bits	90 to 250	10	64	10,000	220 to 440	Industrial- and military-temperature versions available
Exel	XLS2804	4k	512	250 to 450	10	NA	10,000	4.21 (250 nsec)	
	XLS28C16	16k	2k	100 to 250	5	NA	10,000	3.41 (200 nsec)	

4

the necessary programming and erase voltages internally.

You can view the requirement for an external supply of 12V as both a liability and an asset. Certainly it's something extra that you must supply. However, you don't necessarily need or want to design that extra power supply into your product. If you want to allow flash EEPROM programming only under certain, controlled conditions, you can put the 12V supply in a programming station instead of the product itself. A cable can carry the 12V along with the programming signals. This scheme ensures that no inadvertant device programming can occur while your product is in use.

To save transistors, most flash

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	Endurance (write cycles)	Cost ($) (1000)	Comments
Exel (continued)	XLS2864	64k	8k	250 to 450	10	32	10,000	10.88 (250 nsec)	
	XLS28C64	64k	8k	150 to 250	5	64	10,000	10.88 (1500 nsec)	
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	20	NA	10,000	6	
	[illegible]	[illegible]	[illegible]	[illegible]	10	64	10,000	27	
Mosaic	MEM832WJ	256k	32k	90 to 250	10	64	100,000	79 to 168	Military part
	MEM864VX	512k	64k	90 to 250	10	64	100,000	175 to 349	Military part, single-in-line module
	ME128SC	1M	128k	120 to 250	12	64	10,000	560 to 840	Military part
	Puma 2E1000	1M	128k×8, 64k×16, or 32k×32 bits	90 to 250	12	64	100,000	560 to 1480	Military part, pin-grid-array package
NEC	[illegible]	[illegible]	[illegible]	[illegible]	10	NA	100,000	4.50	
	[illegible]	[illegible]	[illegible]	[illegible]	10	32	100,000	7.80	
	[illegible]	[illegible]	[illegible]	[illegible]	10	64	10,000	30	
Samsung	KM28C16	16k	2k	150 to 250	2	32	10,000	2.45	Industrial-temperature version available
	KM28C64	64k	8k	200 to 250	5	32	10,000	4.40	Industrial-temperature version available
	KM28C256	256k	32k	150 to 250	5	64	10,000	23	100,000-cycle and industrial-temperature versions available
Seeq	2816A	[illegible]	2k	[illegible]	10	NA	10,000	5.25 (300 nsec)	2-msec fast-write and 1,000,000-cycle versions available
	[illegible]	[illegible]	8k	[illegible]	10	64	10,000	8.25 (300 nsec)	
	[illegible]	[illegible]	[illegible]	[illegible]	10	64	10,000	30 (300 nsec)	[illegible]-nsec version available
	[illegible]	[illegible]	[illegible]	[illegible]	10	128	10,000	595 (250 nsec)	5-msec fast-write version available
Simtek	STK28C256	256k	32k	70 to 120	10	64	100,000	55 to 80	Military-temperature version available
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	NA	10,000	13.52	
	[illegible]	[illegible]	[illegible]	[illegible]	10	32	10,000	<10	Multiplexed address/data bus, write protection for 1k-byte blocks
	[illegible]	[illegible]	32k	[illegible]	10	64	10,000	29.50	
	[illegible]	[illegible]	[illegible]	[illegible]	10	[illegible]	10,000	100	
	[illegible]	[illegible]	[illegible]	[illegible]	10	256	10,000	275	

Nonvolatile, in-circuit-reprogrammable memories

EEPROMs also lack the circuitry which allows you to erase individual locations. When you erase them, they erase completely. However, you don't always need to erase a flash EEPROM before writing to it. You can always write to an erased, but as-yet-unwritten, location, which results in a write time comparable to that of a full-featured EEPROM. However, once you've written to a location, you must erase most flash EEPROMs entirely before you can rewrite that same location. Thus the flash EEPROM's rewrite time can be very long indeed.

Flash EEPROMs from Catalyst, Philips Components-Signetics, Seeq, and Texas Instruments allow you to erase individual memory sectors and Atmel's flash products (dubbed PEROMs for programmable, erasable ROMs) automatically erase a location before writing new bits into that location. Sector erasure can save your application time

Table 5—Representative flash EEPROMs

Manufacturer	Model	Size (bits)	Organization (bytes)	Read/access time range (nsec)	Max byte write time (msec) (Note 1)	Max erase time (msec) (Note 2)	Sector size (bytes) (Note 3)	Endurance (erase cycles)	Cost ($)	Comments
Advanced Micro Devices	AM28F010	1M	128k	90 to 200	0.4	10,000	NA	10,000	26.50 (100) (200 nsec)	
Atmel	AT29C256	256k	32k	150 to 250	10 (64 bytes)	10	NA	1000	12 to 24 (1000)	Needs only 5V, programs in 64-byte blocks, autoerase during write.
	AT29C010	1M	128k	150 to 250	10 (64 bytes)	10	NA	100	29 to 45 (1000)	Needs only 5V, programs in 64-byte blocks, autoerase during write.
Catalyst	CAT28F512V5	512k	64k	120 to 200	0.4	100,000/ sector	2k	1000	58 (10,000)	Needs only 5V.
	CAT28F512	512k	64k	120 to 200	0.4 (1 or 4 bytes)	10,000	NA	1000	15 (10,000)	Can perform 4-byte writes. 10,000-cycle endurance version available.
	CAT28F010	1M	128k	120 to 200	0.4 (1 or 4 bytes)	10,000	NA	1000	22 (10,000)	Can perform 4-byte writes. 10,000-cycle endurance version available.
Dense-Pac Microsystems	DPZ1MS16P	16M	2M×8 or 1M×16 bits	150 to 250	0.4	30,000	NA	10,000	550 to 750 (1000)	Industrial- and military-temperature versions available.
	DPZ2MS516P	32M	4M×8 or 2M×16 bits	200 to 300	0.4	30,000	NA	10,000	900 to 1200 (1000)	Industrial- and military-temperature versions available.
Intel	28F010	1M	128k	120 to 200	0.4	30,000	NA	10,000	12.05 to 16.95 (1000)	Automotive-temperature version available.
	28F020	2M	256k	150 to 200	0.4	30,000	NA	10,000	32.05 to 35.25 (1000)	
Mitsubishi	M5M28F101P	1M	128k	100 to 150	0.4	10,000	NA	10,000	20 (1000) (100 nsec)	
Philips-Components/Signetics	48F010	1M	128k	200 to 300	11.53	13,425	1k	100	15 (1000)	1000-cycle version available.
Seeq	48F512	512k	64k	200 to 300	11.53	13,425	512	100	13.50 (1000)	1000-cycle version available.
	48F010	1M	128k	200 to 300	11.53	13,425	1k	100	22 (1000)	1000-cycle and military versions available.
SGS-Thomson	M28F256	256k	32k	100 to 200	2.65	840	NA	100	3.50 (10,000)	1000- and 10,000-cycle versions available.
Texas Instruments	TMS29F256	256k	32k	170 to 300	60 (1 to 64 bytes)	150	NA	100	11 (1000)	Needs only 5V. 1000- and 10,000-cycle versions available.
Toshiba	TC58257AP	256k	32k	170 to 250	2.5	2050	NA	100	10.95 to 12.05 (1000)	
	TC58F1001P	1M	128k	150 or 200	2.65	3,000,000	NA	100	15 (100)	10,000-cycle version available.

Notes: 1. A page-write feature allows some flash EEPROMs to program more than one location in the same time required to program one location.
2. For most flash EEPROMs, you must first write zeroes into every memory location to place an equal electrical charge in each memory cell. This pre-erasure operation will greatly extend the time needed to erase large-capacity parts. Most parts will erase in far less time than the maximum listed. Atmel's AT29C256 and AT29C010 automatically erase before writing into a location.
3. You can erase flash EEPROMs with sectors on a sector basis so you need not reprogram the entire part after an erasure.

4

because most flash erasure algorithms require you to write zeros into each location before erasing the part. Each write operation takes hundreds of μsecs or msecs to complete, so writing to the smaller number of locations contained in a sector can mean faster erasure times.

Erase flash EEPROMs carefully

Writing zeros into each memory location charges each of the flash EEPROM's memory cells to the same electrical potential (a charged cell represents a logical zero) so that subsequent erasure will drain an equal amount of free electrons from each cell. Failure to equalize the charge in each flash-EEPROM cell prior to erasure can result in the overerasure of some cells by dislodging bound electrons in the floating gate and driving them out. When a floating gate becomes depleted in this way, the affected transistor can no longer be turned off; overerasure literally destroys some flash EEPROMs. Flash EEPROMs from Seeq are not sensitive to overerasure because the devices' split-gate cell design inherently precludes such problems. As a result, Seeq's flash EEPROM erasure algorithm does not require you to write zeros to all locations before erasing.

Most flash EEPROMs also lack built-in timing circuits so your host CPU must time the programming and erasure algorithms. This feat isn't always as easy as you may think. **Fig 2** illustrates the complexity of Intel's programming algorithm. Flash EEPROMs are relatively new, so the algorithms for programming and erasing the parts vary from vendor to vendor. However, there is an industry-standard identification sequence for flash EEPROMs, so your software can determine what vendor's part it's dealing with if you want to write one piece of code that can program and erase several types of flash memory. When you're implementing the programming and erasure algorithms, you must be careful to plan for inadvertent resets, long delays caused by interrupts, and any other events which may distract the host CPU and disrupt the proper timing of flash operations. Remember, failure to time the erasure properly can permanently damage some flash EEPROMs.

Despite these liabilities, you will still want to consider flash EEPROMs for your nonvolatile memory requirements because they cost less per bit than the other types of nonvolatile, in-circuit-reprogrammable semiconductor memories. The cost for 1M-bit flash memories has dropped by a factor of 10 in the past year and most flash EEPROM vendors state that they will overtake the nonvolatile-memory price leader, the EPROM, in the near future. In fact, some computer manufacturers, such as Airus

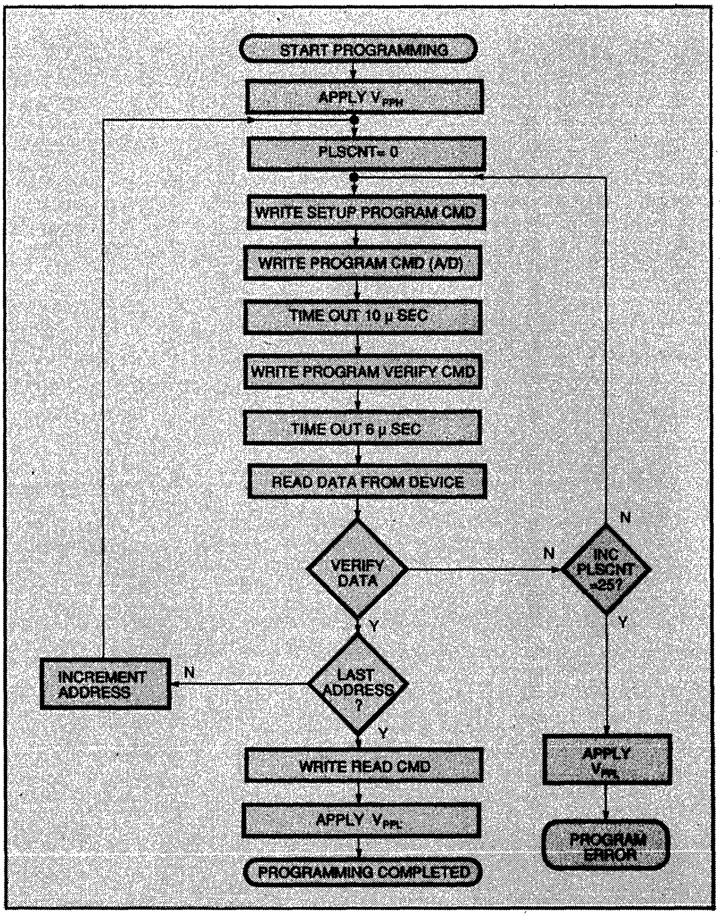


Fig 2—The flash EEPROM's low part cost demands another price: They require a host CPU to execute a fairly complex programming algorithm as illustrated by this programming flowchart for Intel's 28F010 1M-bit flash device.

(Chicago, IL) and Solbourne (Longmont, CO), are already using flash EEPROMs instead of EPROMs to store executable code in their products. The cost advantages of in-circuit reprogramming already outweigh the part-cost differential between EPROMs and EEPROMs if the EPROMs must be changed just once. The labor cost incurred by opening a product and changing the EPROMs tilts the balance in favor of flash memory.

In addition, flash EEPROMs allow you to eliminate the IC socket that you often use with EPROMs just in case you need to change your code. Elimination of the socket further reduces the parts-cost difference between EPROMs and flash EEPROMs and boosts system reliability because sockets can be one of the least reliable parts of a system. You can solder flash EEPROMs directly onto your circuit board and still accommodate code changes through in-circuit reprogramming.

EPROM replacement represents a big market for flash EEPROMs because all μP-based and many microcontroller-based systems contain at least a few EPROMs that flash devices can replace one for one. However, flash EEPROM vendors, such as Intel and Toshiba, have set their sights on an even bigger market. They will eventually replace a large number of disk drives with flash EEPROMs as well. At first glance, that goal seems more like an impossible dream than a realistic marketing plan. Disk drives are well established as the storage medium of choice for a range of computer systems. So was core memory before semiconductor RAM appeared. No technology in the electronics industry lives forever.

To overtake disk drives, flash-memory vendors must first overcome a few obstacles. For example, flash EEPROMs do not exactly match the characteristics and capabilities of disk drives so they cannot exactly replace them. Read/access times are much faster for flash devices than for disk drives, which is a real advantage. Storage times can also be much faster for flash EEPROMs, but they can also be much slower if you need to erase them before writing new information. In addition, flash EEPROM vendors aren't yet ready to match the low cost per bit you achieve with disk drives. However, they are preparing for the battle.

As part of that preparation, Intel recently introduced two plug-in memory cards based on its flash EEPROMs. The company's 1M-byte iMC001FLKA and 4M-byte iMC004FLKA cards measure 85.6 × 54 × 3.3 mm and conform to the 68-pin specification developed by the Personal Computer Memory Card International Association. Because memory cards do not work like disk drives, Intel has worked with Microsoft (Redmond, WA) to develop a Flash File System that allows a DOS-based computer to store and retrieve files from Intel's memory cards. The 1M- and 4M-byte cards sell for $298 and $1198 (1000), respectively. A developers' kit containing a 1M-byte memory card, an IBM PC interface board, and an evaluation copy of Microsoft's Flash File System costs $499.95. At today's prices, flash EEPROMs clearly do not threaten disk drives for most applications. However, if your design must meet rugged or low-power requirements, these memory cards give you a viable alternative to consider.

As usual, IC technology refuses to stand still. In the past year, flash EEPROM prices dropped by an order of magnitude. That change occurred without any increase in device density. As flash EEPROM vendors start doubling and redoubling device capacities, they plan to drop the cost-per-bit of their products until they compete with disk drives. If this plan still seems overly ambitious to you, consider this: More than one IC vendor plans to make the flash EEPROM its IC technology driver in the 1990s.

EDN